

HINS Front End Superconducting Focusing Solenoids A Tool for Modeling Quench Propagation and Related Protection Issues

S. Obraztsov, I. Terechkine

1. Introduction

The work was done in an attempt to make possible formal and reliable analysis of quench-related properties of focusing solenoid of HINS Front End.

The name “solenoid” will refer to a system consisting of three coils: one main coil and two bucking coils. In each coil propagation of a quench initiated in some point can be predicted taking into the account the magnetic field this coil sees. General layout of a solenoid is shown in Fig. 1. Only relevant details are shown here; better understanding of the design can be obtained from [1].

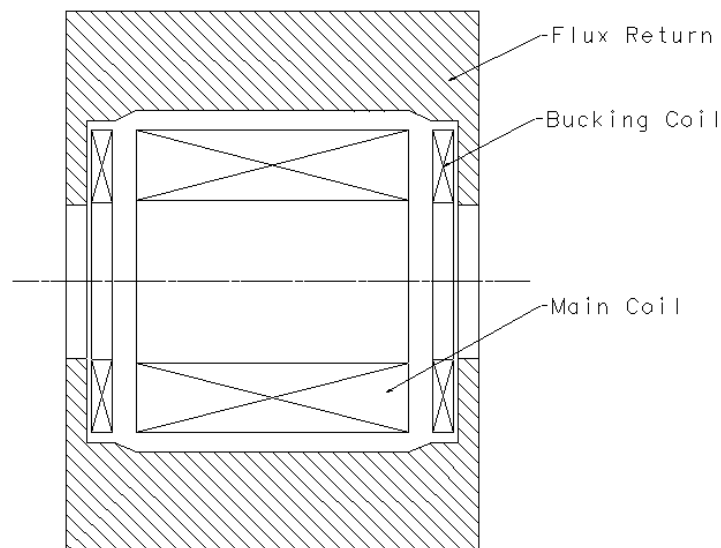


Fig. 1: General layout of a focusing solenoid

The software is to model quench propagation in any coil of a solenoid in the magnetic field environment this coil sees. The initial point of the quench can be chosen arbitrarily in the coil that is being analyzed. The quantities related to the quench propagation are:

- Temperature at any point inside the coil at any time;
- Critical current at any point at any time;
- “Resistivity” (resistance of one meter of the strand) of the coil winding at any point at any time;
- Coil resistance at any time;
- Voltage to ground in the coil for any layer at any time;
- Current profile after quench, taking into the account parameters of the external circuit;
- Other properties that can be derived from the listed above.

We are mainly dealing here with solenoids for the HINS Front End Linac because certain features of the program are developed to simplify modeling of these solenoids (like epoxy impregnation or insulating system). These features are not principal though and can be easily changes to allow a broader range of the devices to be analyzed.

Some features, like a delay of switching to a protection mode or user's interface, have not been implemented, although they can be added later.

The goal of the first stage of this work was to provide means for analyzing quench behavior of the PD focusing solenoids. This information will be needed when a quench protection system will be designed for solenoids installed in the beam channel of the front end of PD (totally ~ 50 of them).

2. Approach to modeling and a block diagram

Given a general layout (Fig. 1) of the solenoid and results of magnetic modeling, a map of magnetic field, defined by the excitation current, can be generated for any coil in the assembly. If a local temperature rise above the critical point occurs in a coil, the temperature of the corresponding turn will further rise and heat will transfer to neighboring areas through thermal conductance. The rate of the temperature rise will be defined by heat deposition rate through resistive losses and by thermal conductivity and heat capacity of the winding, which can differ in the radial and longitudinal directions. Knowing the critical surface of a superconductor used to make the coils, one can find a moment for any turn when it becomes normal. To simplify the task, only a 2D problem is solved, so the heat transfer along the strand is neglected. As it was shown in [2], this heat transfer, although playing certain role, is significantly less than the transverse heat propagation if the coil is impregnated with epoxy or another media with good thermal conductivity, as it is in our case. So, it is assumed that the whole turn becomes normal to start the process, and the propagation is only in the transverse directions (radial and longitudinal).

The modeling is performed by sequential stepping in time and recalculation of the next quantities for each turn and each step: heat deposition, temperature rise, critical current, and resistance. After each step, the sum of resistances of all the turns gives the updated value of the coil resistance. Knowing the inductance and resistance of the coil and the circuit (there are three coils in the circuit and there is an external resistance), the current drop can be calculated and the value of the current updated. This results in the updated magnetic field map and a new steps repeat the process until the current becomes close to zero. The block diagram of the calculation process is summarized in Fig. 2.

Input M-file file contains all the parameters that are needed for the calculation: geometrical, electrical, computational, etc. In this file calculation of coil's electrical properties, like inductances, is performed based on its geometrical properties. An effective rectangular turn cross-section shape and an effective inter-layer and inter-turn insulation thickness are also defined in this file.

Magnetic field transfer function B/I must be found outside of the program and presented by a rectangular array of arbitrary (but known) size. This size and the number of layers in the coil are used to make an interpolation fit of the magnetic field to every turn in the coil.

The output file contains scripts of possible outputs of the program. A user is free to add other scripts. Most of the M-files used by the program will be discussed in more details later in this manual.

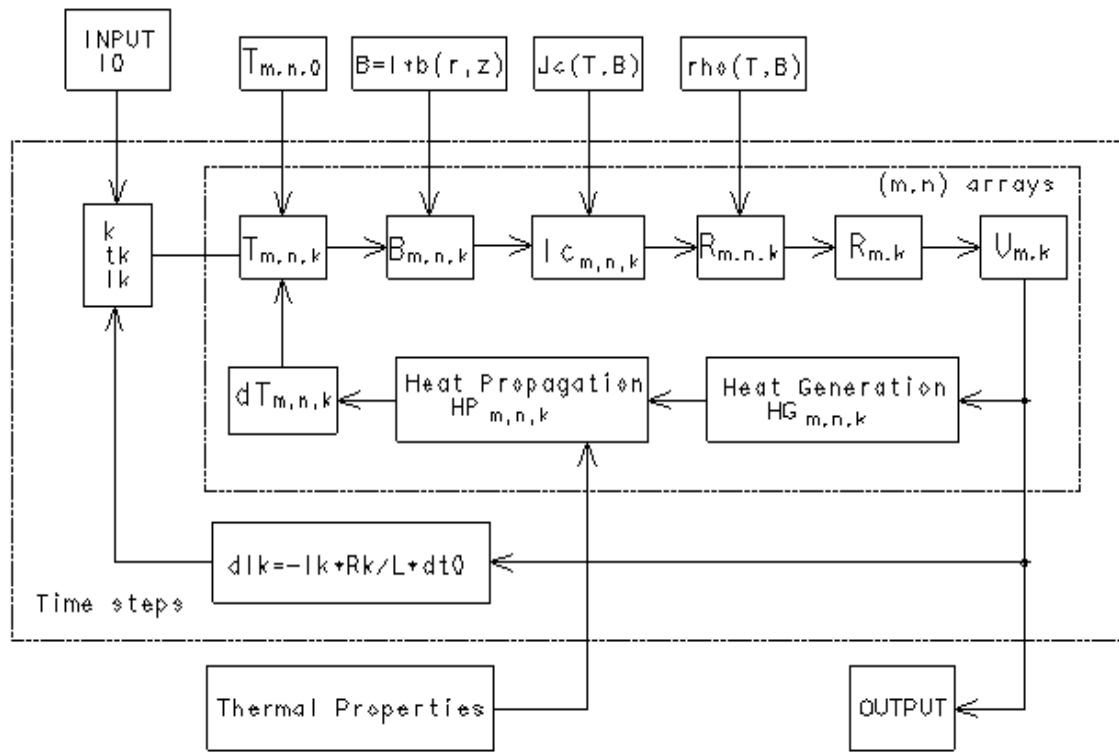


Fig. 2: Block diagram of the program

The next parts are devoted to description of several M-files that are used to model the problem.

3. Material Properties

One of key issues during the modeling is using correct thermal characteristics of the involved materials, namely heat capacity, thermal conductivity, and electrical conductivity, in the temperature range of interest. These characteristics change by orders of magnitude as the temperature changes from 4 K to 300 K. Depending on the form these characteristics are available, a convenient functions are found for parameterization or interpolation is used to find the needed quantities.

3.1. Specific Heat

Specific heat data for all the materials involved in heat transfer are specified in the M-file [SpecificHeat.m](#). These materials are Cu, NbTi, and G10 (which models epoxy-impregnated fiberglass tape used for interlayer insulation). If needed, more material can be added in the file. Corresponding graphs are shown below. They are based on the analytical approximations of the data available from NBS [3] and other relevant databases.

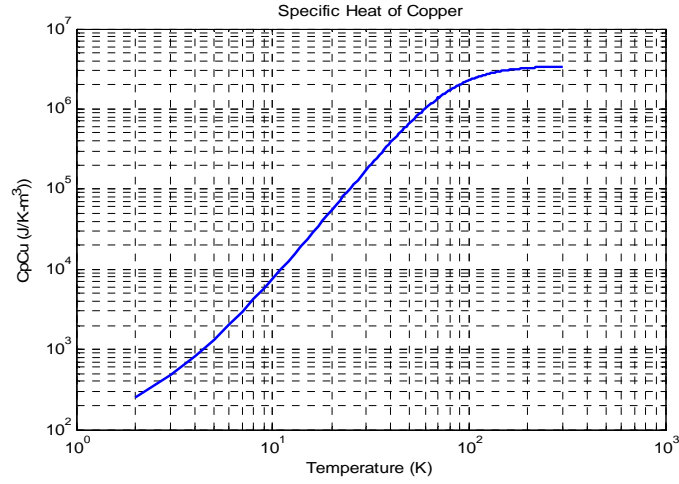


Fig. 3: Specific Heat of Copper

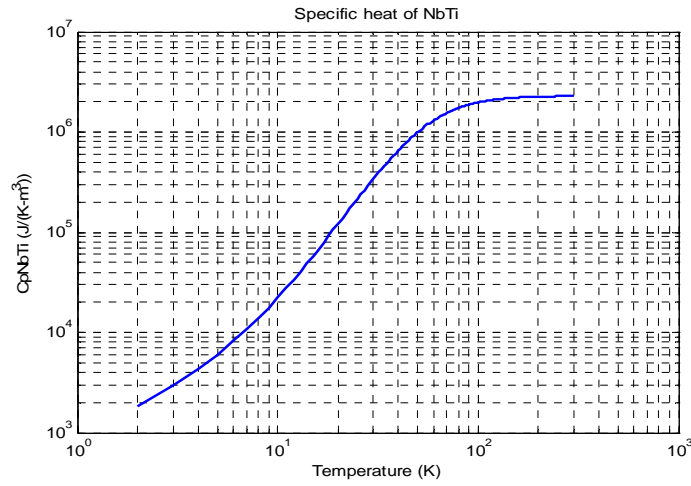


Fig. 4: Specific Heat of NbTi

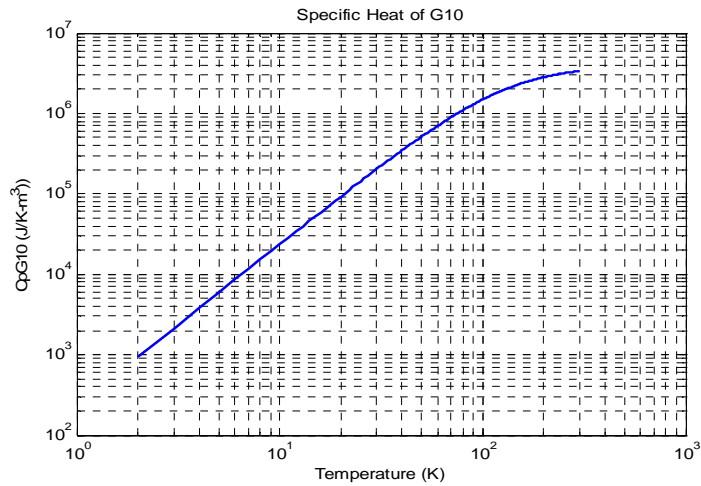


Fig. 5: Specific Heat of G-10

It is worth to pay attention to the fact that the values of the Specific Heat for the listed material change in the range spanning 3 – 4 orders of magnitude.

3.2. Thermal Conductivity

Thermal Conductivity data for all the materials involved in heat transfer are specified in the M-file [ThermalConductivity.m](#). These materials are Cu and G10. NbTi embedded in the copper matrix of the strand provides little bypass for the heat transfer in the strand, so this material was not taken into the account. If needed, more material can be added in the file. Corresponding graphs are shown below. They are based on the interpolation of the data tables extracted from [3] and [4]. 99.98% pure copper was used to represent the data. This corresponds to the RRR values of ~ 50 .

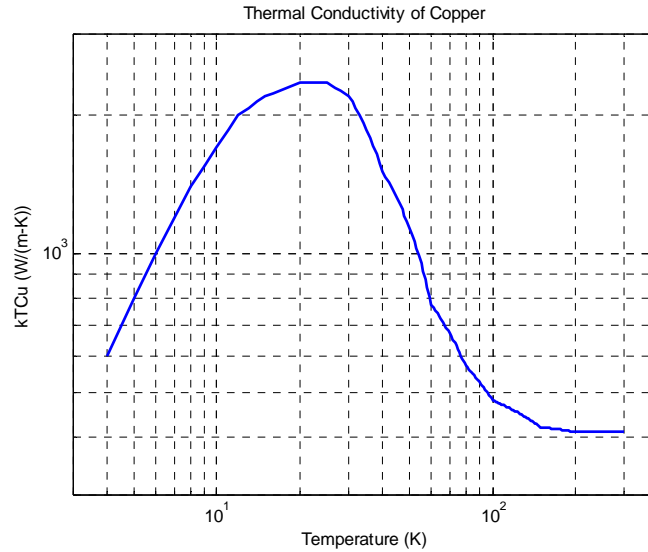


Fig. 6: Thermal Conductivity of Copper

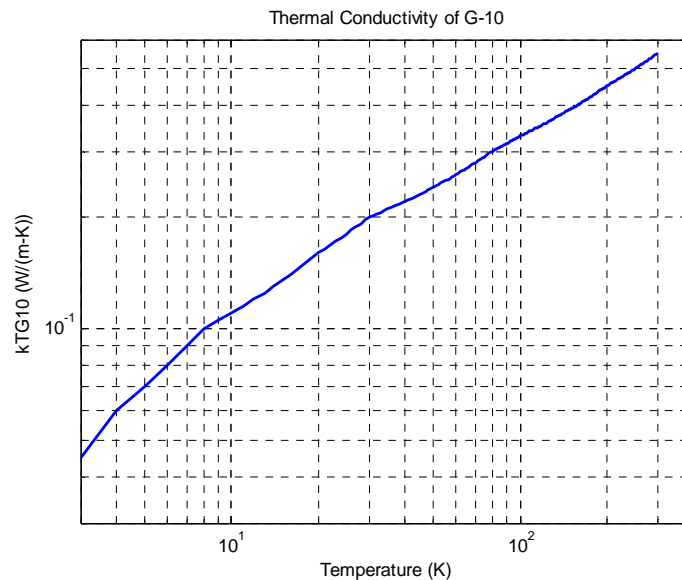


Fig. 7: Thermal Conductivity of G-10

The thermal conductivity changes by about one order of magnitude for both materials, although being quite different from each other.

3.3 Specific Resistance

The specific resistance of Copper depends not only of the temperature, but also on the purity of the material and on the level of magnetic field. The M-file [SpecificResistance.m](#) uses analytical approximation known from [5]. Example graph in Fig. 8 provides curves for several values of the magnetic field and $RRR = 50$.

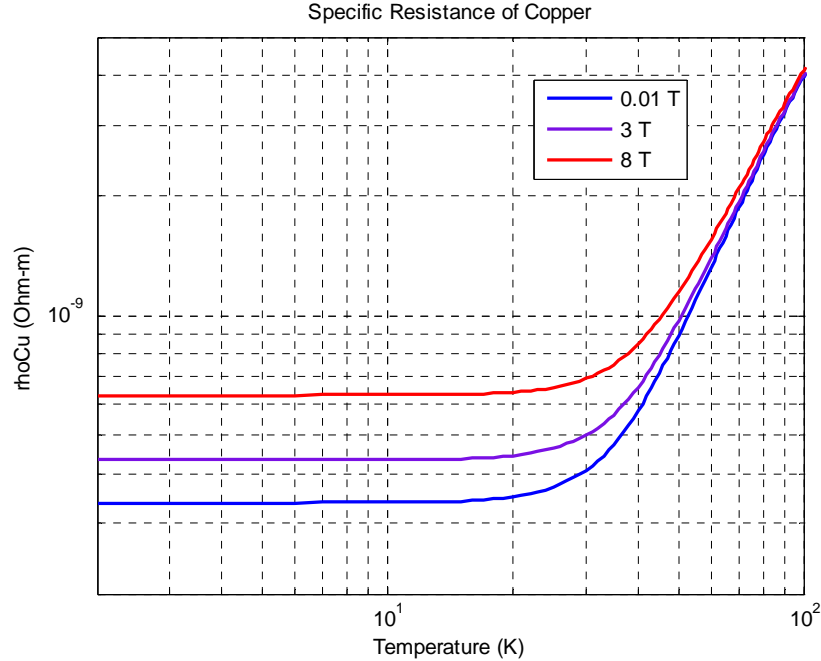


Fig. 8: Specific resistance of Copper ($RRR = 50$) at different levels of the magnetic field.

At low temperature, specific resistance of Cu does not depend on the temperature, but depends on the magnetic field. As the temperature increases, the resistance becomes more dependent on it, but less dependent of the magnetic field. So, both the magnetic field and the temperature must be taken into the account while calculating resistance of the coil.

3.4. Critical Surface of NbTi

Critical current density in NbTi depends on the temperature and the level of the magnetic field. It was difficult to find an approximation that worked in the full range of temperatures and magnetic field. So, the analytical approximation was used [6] that works well in the upper interval of the field and it was extrapolated to the lower field. This approximation has been implemented in the form of a M-function $J_c(\mathbf{B}, \mathbf{T})$. The resultant critical surface is shown in Fig. 9. Although not precise, it works well to serve our needs. If needed, other parameterization can be used instead.

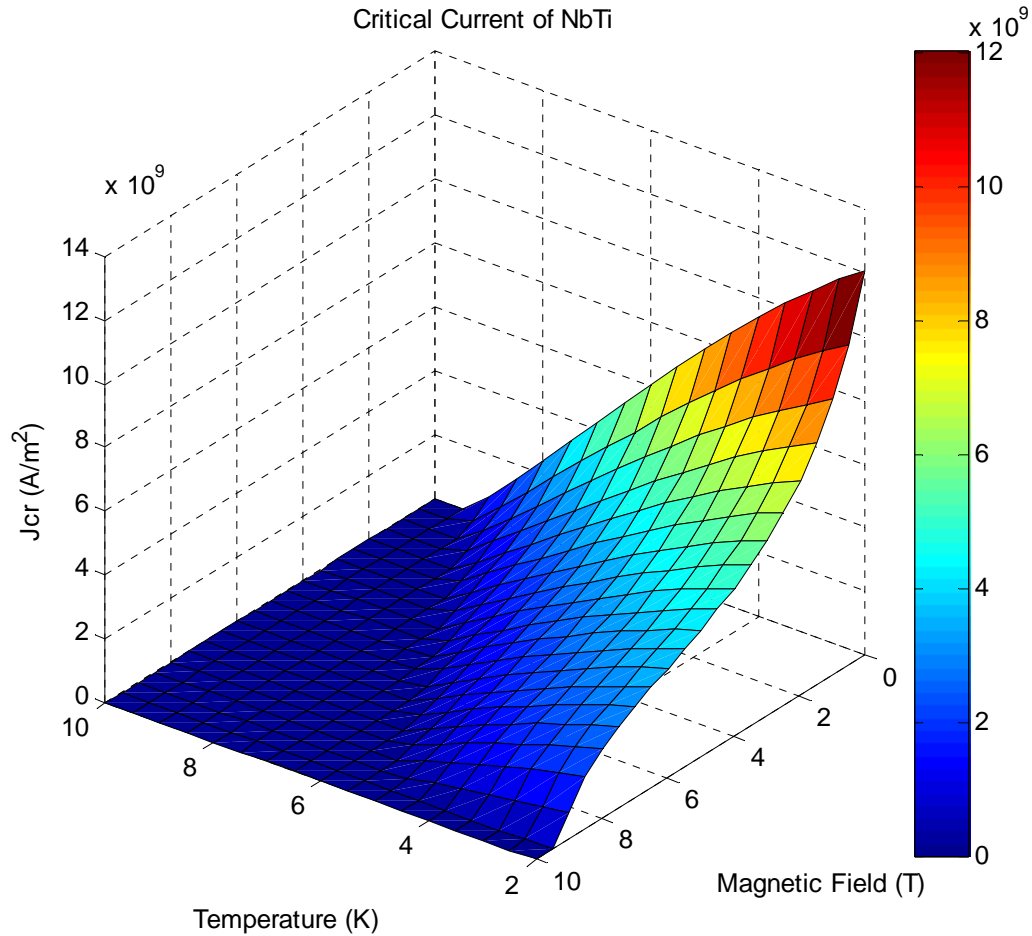


Fig. 9: Critical Current Density of NbTi

4. Input.m

There are several parts in this M-file, each addressing certain class of parameters: Geometrical, Electrical, Material, Initialization, Computational, Transfer Function Array, and Output.

4.1. Geometrical Parameters

Although some of the geometrical parameters do not require comments, like the coil inner **Di** and outer **Do** diameters, coil length **L**, strand diameter **d** and its cross-section **Sb**, number of turns **Nt** and layers **Nl**, others require some explanations. Although a round strand is used for coil fabrication, in the program an equivalent rectangular strand is used. Cross-section area of this strand must be the same as the virgin strand has. The number of the layers in the coil and the total number of turns also must be the same. Then the height and the width of the equivalent strand can be calculated if one knows a ratio “**rat**” between the thickness of the interlayer insulation and the insulation thickness between the turns in each layer. This ratio depends on winding technique and insulation material used and usually is in the range between 0.5 and 2. Assuming a certain value for

this ratio, width **w** and height **h** of the strand and also interlayer insulation thickness **tlay** and turn insulation thickness **tturn** can be found. To take into account different insulating properties near the boundary of a coil, the next values can also be provided: the ground insulation on the spool barrel **tsp**, the ground insulation at the spool flanges **tfl**, and the insulation thickness above the outer layer on the coil **tout**.

4.2. Electrical Parameters

In this section of the input file, the external circuit resistance and inductance are provided or found. In the case of a focusing solenoid, while modeling quench propagation in one of the coils, one needs to take into the account the inductances of two other coils. Outer circuit resistance is usually presented in a form of a dump resistance. No time delay options are introduced at this moment, although this can be done. The inductance of the coil can be calculated based on its geometrical properties, but it was found convenient to calculate the so-called “reactance” of each layer **React_Layer**. For each layer, its reactance provides a ratio between an inductive voltage across the layer and the current rise rate. It differs from just layer inductance by taking into account also mutual inductances between any two layers in the coil. Knowing the reactance array, it is easy to calculate the coil inductance just by summing through all the layers. The resultant coil inductance is quite close to what can be calculated directly using the coil geometrical properties and handbook expressions for inductance.

4.3. Material-Related Strand Properties

This section does not require much of an explanation. Parameter **k_Cu** is relative volume of copper in the strand, and **k_nCu** is a relative volume of NbTi in the strand. Both of these parameters can be calculated if the Cu-to-nCu ratio of a strand is known.

4.4. Quench Initialization

In this section, one must make input of the location of the initial place (the layer number **l0** and the position of the turn with elevated temperature in the layer **t0**) of quench and the temperature **Tin** of this turn (above the critical temperature). Also the initial coil current **I0** and the current derivative **dIdt_0** are introduced. The array of the initial temperature distribution through the coil **Tturn** is initialized and the LHe bath temperature is assigned.

4.5. Computational Properties

These properties include a time step value **stepT**, which affects a precision of the modeling, and a number of steps **Nsteps**, that reflects the total time of the process to model.

4.6. Transfer Function Array

To simplify input of the distribution of the magnetic field through the coil volume, a provision is made to use an arbitrary array of magnetic field as an input and then recalculate magnetic field to correspond to the position of each turn. This way one does not need to make input of large number of data points.

The initial array of magnetic field is introduced in the M-file “[MagneticField.m](#)”. It is obtained in the next way:

- Using any magnetic modeling code, data is taken on a grid covering the coil cross-section. Coordinate $Z = 0$ is accepted on the left border of the coil; then coordinate **Lmax** will correspond to the right border. Coordinate **Rin** corresponds to the top raw in the lower half of the coil cross-section (Fig. 1); coordinate **Rout** corresponds to the bottom raw of this cross-section. The size of the grid will affect slightly precision of modeling, but this is only a second order effect.
- For the main coil, the data can be taken only for one half of the coil and then reflected against the median plane using a spreadsheet table.
- In an array **MagF**, data along the length of the solenoid (in Gauss) can be read horizontally, starting from the left; the data corresponding to different layers are arranged vertically, starting from the inner layer on the top of the table.
- The current at which the data are taken is written as **Icoil**.

The intermediate transfer function **F** is obtained by dividing **MagF** by **Icoil**.

To generate a transfer function corresponding to position of any turn in the coil, simple interpolation is used that distributes the data in **F** onto every of the turns, resulting in the array **B_Icoil** with the size (**Nl**, **Nt**).

4.7. Output Parameters

As it was told earlier, this file serves as a complimentary mean to save time on making requests of certain output graphs. It can be modified or changed to customize the output during study of particular systems.

When trying to visualize transient behavior of the output, the whole number of steps **Nsteps** is divided by the number of slices **Nslice**. The output data are saved in the beginning and in the end of every slice forming arrays detailing time-dependent properties of the quenching system. These arrays can further be analyzed as needed.

5. Main.m

This executable script controls the modeling process. First it activates the M-files with material properties: [SpecificResistance](#), [ThermalConductivity](#), and [SpecificHeat](#), so that these functions are available when needed. Next, the initial conditions are assigned to current **It = I0** and current derivative **dIdt = dIdt_0** and array variables are initialized.

These variables are:

Icr – a two-dimensional array of critical currents at the turns' location;

Rturn1 – a two-dimensional array of resistivity at the turns' location;

Rlayer – a one-dimensional array of layers' resistance;

Rcoil – a scalar of the coil resistance;

dU – a one-dimensional array of layers' voltage gain;

U – a one-dimensional array of layers' voltage;

Time – a one-dimensional array of time moments when the output data are stored;

Icoil_time – a one-dimensional array of the coil current at the moments defined by the array **Time**;

dIdt_time – a one-dimensional array of the coil current derivative at the moments defined by the array **Time**;

MaxT_time – a one-dimensional array of maximal temperatures inside the coil at the moments defined by **Time**;

Rcoil_time - a one-dimensional array of the coil resistances at the moments defined by **Time**;

U_time - a two-dimensional array of the coil layers' voltages at the moments defined by **Time**;

The time loop starts with recalculation of the magnetic field from the arbitrary mesh to the location of the turns. Knowing the magnetic field and the current allows finding the critical current **Icr** at the location of each turn of the coil by using the function **Jc(B,T)** and parameters of strand. By comparing the critical current at the location of each turn with the coil current, one can find whether this turn is superconducting or normal. If it is normal, the turn resistivity **Rturn1** is calculated taking into the account known specific resistance at given temperature and magnetic field. By summing the resistances of turns within one layer, an array of the layer resistance **Rlayer** is updated. By further summing the resistances of the layers, the coil resistance is **Rcoil** found.

Now, as we know the resistance and reactance of each layer, the voltage gain **dU** and the voltage to ground **U** can be found.

Because we know at the same moment the current and the resistances of each turn, we can calculate for each turn the energy deposition during one time step and corresponding temperature growth, as we know specific heat (as a function of temperature) for each turn and the initial temperature field in the coil. Also, because the thermal conductivity is also known, we can apply some logic to calculate heat dissipation (or acquisition) through thermal conductivity and translate it into corresponding temperature rise (or drop). As a result, an updated temperature array will replace the initial array **Tturn**.

Before the first time step is completed, the current time is compared with what is written in the array **Time** and based on results of this comparison relevant data can be written into the arrays that define transition properties of the system: **Icoil_time**, **dIdt_time**, **MaxT_time**, **Rcoil_time**, and **U_time**.

After this comparison is made, the current time derivative and the current are recalculated based on the updated values of the coil resistance. The time cycle is then repeated with the updated arrays of major coil properties.

After the final time step is made, the output file is called to produce desired graphs and tables. This output can be tuned in accordance with the needs of the modeling.

6. Program verification

The work of the program was checked by a comparison of obtained results for a test coil with those obtained by other means, including direct observation of the coil behavior. In [2] and [7] a method was developed of quench propagation analysis in solenoids that used MathCad as a computational engine. This method made use of sequential analytical representation of the processes during quench propagation. Although it required a lot of fitting work, it allowed getting results relatively fast, and was appropriate for analysis of a few test solenoids that were fabricated. This section compares results obtained in [2] and [7] with those obtained by using the developed program applied to the solenoid of the identical geometry. The properties of interest are the current time profiles corresponding

to different initial values of the current, maximum temperature in the coil, and maximal voltage. Fig. 10 shows the current decay time profiles at the initial current settings similar to used in [2] and [7]. This graph must be compared with the graph in Fig. 19a in [7] to conclude that these two graphs are very close to each other. As in [2], the initial quench point was located at the point of the maximum field.

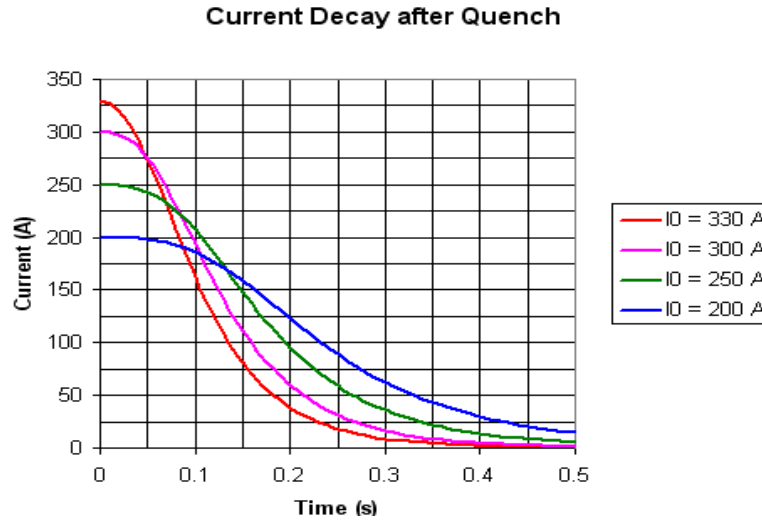


Fig. 10: Current decay profile in the test coil

In [7], maximal temperature of the coil reaches ~ 70 K at $I_0 = 330$ A and the point of the maximum temperature is located at the point of the maximum field. Using the program, we get the graph shown in Fig. 11.

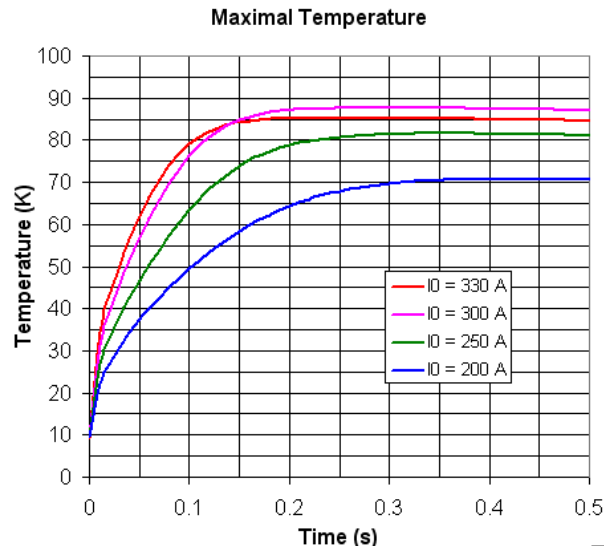


Fig. 11: Maximum temperature time profile in the test coil

Here we see that the maximal temperature at $I_0 = 330$ A is ~ 85 K. The temperature at $I_0 = 300$ A is a bit higher. Slight drop of the temperature after reaching the maximum is due to LHe cooling, which was not taken into the account in [2] and [7]. The increase of the temperature at 300 A is probably due to lower quench propagation velocity, that is not fully compensated by the lower stored energy in the coil.

The analysis of the voltage in the coil attempted in [7] proved quite inaccurate because of the intrinsic noise introduced by the fitting process. It indicated (see Fig. 24 in [7]) that the maximal voltage is, most probably, in the middle of the coil, that this value is of the order of 15 V, but it did not converge properly to give zero voltage at the outer layer, as it must be by definition. The reason of uncertainties in that case was uncertainty in the current time profile introduced by parameterization. Similar simulation made with the use of the numerical algorithm gave more detailed and consistent picture, shown in Fig. 12.

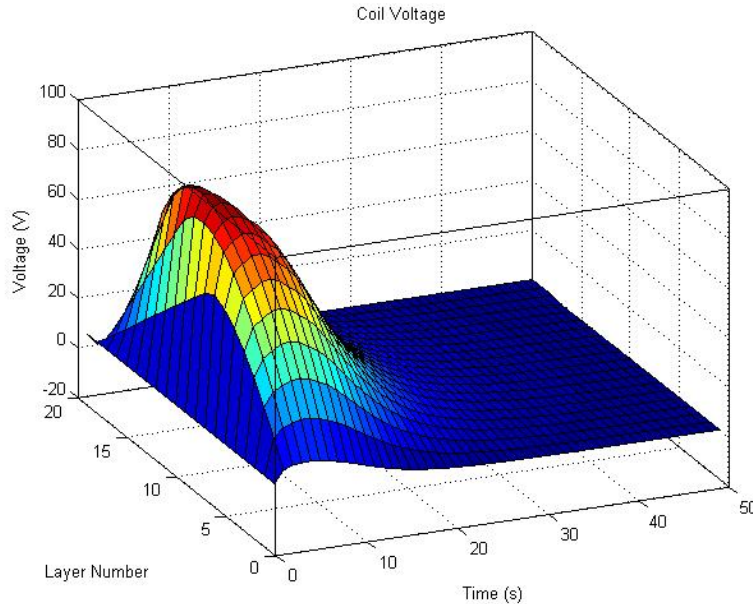


Fig. 12: Voltage distribution time profile in the test coil. The readings of the time scale must be divided by 100 to get result in seconds.

The maximal voltage reaches ~60 V at the moment ~ 0.1 s, in the middle of the coil (layer 10). The outer layer (#20) has zero voltage to ground. In [7] an explanation was provided to the uncertainty of modeling of the voltage distribution. The program gives more consistent results, and we need to take into the account much higher level of the voltage in the coil.

Another fact that somehow proves that the results obtained by the software have some sense is that when the current time profile obtained with the use of the program is compared with the current decay measurements [8], the two curves are quite close to each other. The current decay is a reflection of the thermal processes inside the coil, so this points towards the conclusion that there are no big flaws in the software.

7. Conclusion

Although probably not perfect and not equipped with a convenient user's interface yet, the developed software allows thorough study of quench development in the coils of the focusing solenoid and other similar devices, providing valuable input information for the linac quench protection system design. Typical time of resolving one data series (0.5

seconds with a step of 10^{-4} s, totally 5000 time steps) for the test solenoid (20 layers and 110 turns in a layer) takes from 2 to 5 hours depending on what computer was used. For sure, some optimization can be applied to improve the performance, which has not been done yet.

The software can be obtained free on request.

Future work in this direction will include study of the solenoids designed for the HINS linac front end, and some improvements to the program that would simplify and, may be, visualize input.

References:

1. G. Davis, et al, Linac CH-Type Cavity Section Focusing Solenoid Cold Mass Design, TD-06-020, FNAL TD, 2006
2. I. Terechkine, P. Bauer, PD Front End Focusing Solenoid Quench Protection Studies. Pt I: Method Description and the First Iteration, TD-06-003, FNAL TD, 2006
3. A Compendium of the Properties of Materials at Low Temperature. Wadd Technical Report 60-56, Pt II, Oct 1960.
4. Yukikazu Iwasa, Case Studies in Superconducting Magnets, Plenum Press, N-Y, 1994, p. 385
5. P. Bauer, Stability of Superconducting Strands for Accelerator Magnets, Dissertation, CERN, 1998.
6. Arnaud Devred, Review of Superconducting Dipole and Quadrupole Magnets for Particle Accelerators, DAFNIA/STCM 98-07, August 1998.
7. I. Terechkine, PD Front End Focusing Solenoid Quench Protection Studies. Pt II: Test Solenoid Quench Protection, TD-06-004, FNAL TD, 2006